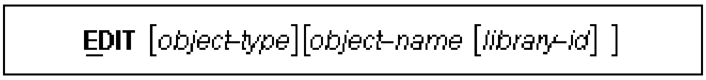
EDIT

- For Mainframes
- For UNIX/OpenVMS
- For Windows

EDIT for Mainframes



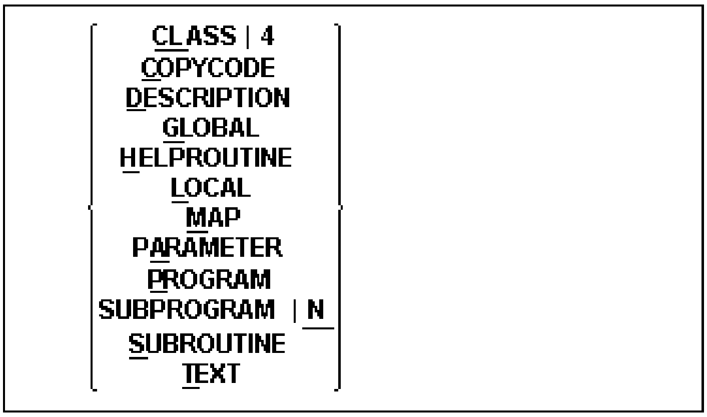
```
EDIT [object-type][object-name [library-id] ]
```

The EDIT command is used to invoke a Natural editor for the purpose of editing the source form of a Natural programming object.

Below is information on:

- object-type
- object-name
- library-id

object-type



```
CLASS | 4
COPYCODE
DESCRIPTION
GLOBAL
HELPROUTINE
LOCAL
MAP
PARAMETER
PROGRAM
SUBPROGRAM | N
SUBROUTINE
TEXT
```

Which editor is invoked depends on the type of object to be edited:

- If the object to be edited is a LOCAL data area, GLOBAL data area or PARAMETER data area, the data area editor will be invoked.
- If the object to be edited is a MAP, the map editor will be invoked.
- All other object types - CLASS | 4, PROGRAM, SUBPROGRAM, SUBROUTINE, HELPROUTINE, COPYCODE, TEXT, DESCRIPTION - are edited in the program editor. (A DESCRIPTION is a program description as stored and maintained in the Predict Data Dictionary; an object of this type can only be edited if Predict is installed.)

If you specify the name of the object you wish to edit, you need not specify its object type.

object-name

With the EDIT command, you specify the name of the object you wish to edit (maximum 8 characters); Natural will then load the object into the edit work area of the appropriate editor and set the object name for a subsequent SAVE, CATALOG, STOW command.

If you do not specify an *object-name* and there is no object in the edit work area, the empty program editor screen will be invoked where you can create a program.

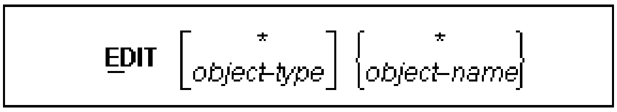
For EDIT DESCRIPTION, the *object-name* must be the name as defined as a Natural member in the Predict program definition.

library-id

If the object you wish to edit is not contained in the library you are currently logged on to, you must specify the *library-id* of the library in which the object to be edited is contained.

The setting for *library-id* must not begin with "SYS" (except "SYSTEM").

If Natural Security is active, a *library-id* must not be specified, which means that you can only edit objects which are in your current library.



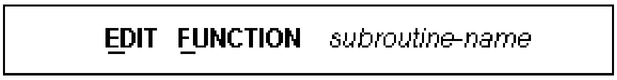
```
EDIT [     *       ] {      *      }
     [ object-type ] { object-name }
```

If you do not remember the name of the object you wish to edit, you can use this form of the EDIT command to display a list of objects, and then select from the list the desired object.

EDIT * displays a list of all objects in your current library.

EDIT *object-type* ***** displays a list of all objects of that type in your current library.

To select an object from a certain range of objects, you can use asterisk notation and wildcard notation for the *object-name* in the same manner as described for the system command LIST.



```
EDIT FUNCTION subroutine-name
```

The EDIT FUNCTION command may be used to edit a subroutine using the subroutine name (not the object name).

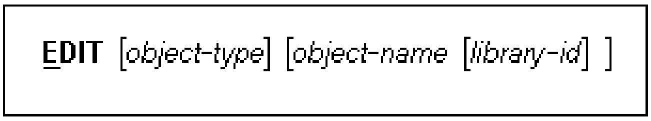
Example:

```
DEFINE SUBROUTINE ABC
  ...
  END-SUBROUTINE
  END
```

Assuming that the above subroutine has been saved under the object name "SUB", you may edit subroutine ABC either by issuing the command EDIT S SUB or by EDIT F ABC.

EDIT for UNIX/OpenVMS

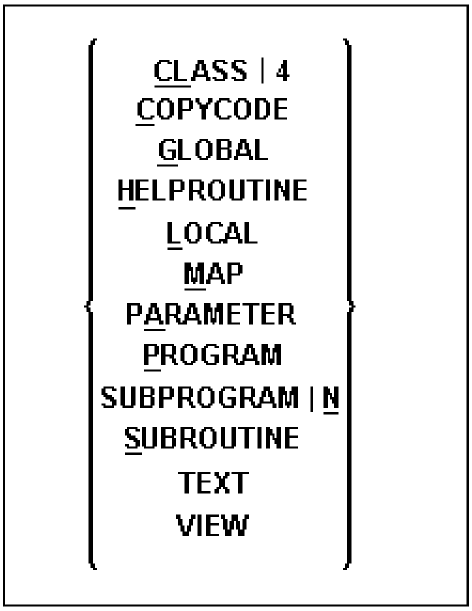
Syntax 1



```
EDIT [object-type] [object-name [library-id] ]
```

The EDIT command is used to invoke a Natural editor for the purpose of editing a Natural object.

object-type



```
CLASS | 4
COPYCODE
GLOBAL
HELPROUTINE
LOCAL
MAP
PARAMETER
PROGRAM
SUBPROGRAM | N
SUBROUTINE
TEXT
VIEW
```

The EDIT VIEW only works in the current library and when an *object-name* is specified.

If you are in the program editor and enter EDIT object-name (whereas the Natural object is of type copycode, program, subprogram,subroutine, class, text, helproutine), a parallel edit session will be opened. See also the Program Editor, editor command NEXT.

object-name

As *object-name* you specify the name of the object to be edited (maximum 8 characters; exception: view names contain maximum 32 characters); Natural will then load the object into the edit work area of the appropriate editor and set the object name for a subsequent SAVE, CATALOG, or STOW command.

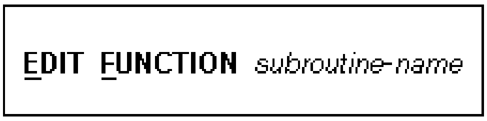
If you do not specify an *object-name* and there is no object in the edit work area, the empty program editor screen will be invoked where you can create a program.

library-id

As *library-id*, you specify the library in which the object to be edited is contained. The *library-id* need only be specified if the object to be edited is contained in a library other than the one you are currently logged on to.

Under Natural Security, you cannot specify a *library-id*; that is, you can edit only objects which are stored in your current library.

Syntax 2



EDIT FUNCTION *subroutine-name*

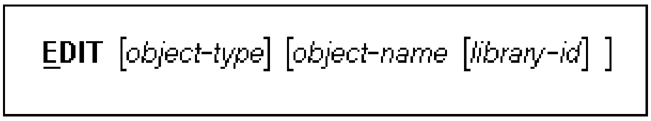
The EDIT FUNCTION command may be used to edit a subroutine using the subroutine name (not the object name).

Example:

```
   DEFINE SUBROUTINE ABC
   ...
   END-SUBROUTINE
   END
```

Assuming that the above subroutine has been saved under the object name "SUB", you may edit subroutine ABC either by issuing the command EDIT S SUB or by EDIT F ABC.

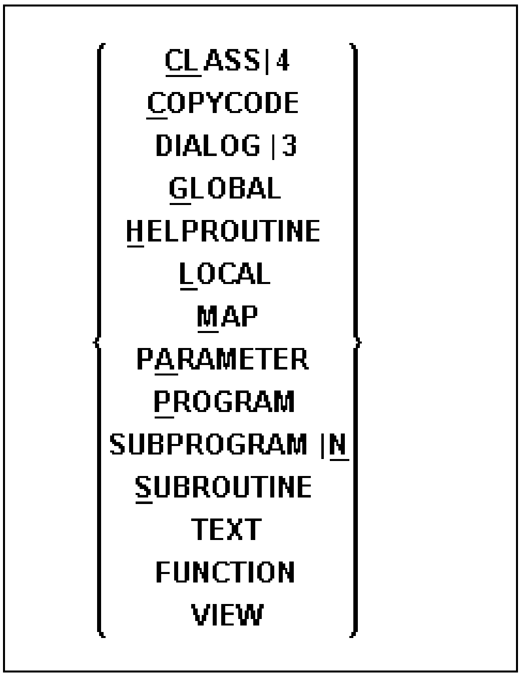
EDIT for Windows



```
EDIT [object-type] [object-name [library-id] ]
```

The EDIT command is used to invoke a Natural editor for the purpose of editing a Natural object.

object-type



```
{ CLASS|4          }
{ COPYCODE         }
{ DIALOG |3        }
{ GLOBAL           }
{ HELPROUTINE      }
{ LOCAL            }
{ MAP              }
{ PARAMETER        }
{ PROGRAM          }
{ SUBPROGRAM |N    }
{ SUBROUTINE       }
{ TEXT             }
{ FUNCTION         }
{ VIEW             }
```

As *object-type* you specify the type of object to be edited.

object-name

As *object-name* you specify the name of the object to be edited (maximum of 8 characters); Natural will then load the object into the edit work area of the appropriate editor and set the object name for a subsequent SAVE, CATALOG, or STOW command.

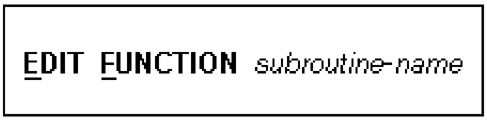
If you enter the EDIT command without any parameters, the objects marked in the "Objects" or "DDMs" window are loaded into the appropriate editors.

library-id

As *library-id*, you specify the library in which the object to be edited is contained. The *library-id* need only be specified if the object to be edited is contained in a library other than the one you are currently logged on to.

Under Natural Security, you cannot specify a *library-id*; that is, you can edit only objects that are stored in your current library.

Edit Function



EDIT FUNCTION *subroutine-name*

The EDIT FUNCTION command can be used to edit a subroutine using the subroutine name (not the object name).

Example:

```
   DEFINE SUBROUTINE ABC
   ...
   END-SUBROUTINE
   END
```

Assuming that the above subroutine has been saved under the object name "SUB", you can edit subroutine ABC by issuing either the command EDIT S SUB or the command EDIT F ABC.